[0090] As shown in **FIG. 6D**, the features **184** are classified, i.e., each finger/thumb is identified, and grouped for each of the images **180** and **182**. In this particular case, the features **184**A associated with the knob **170** are grouped together to form group **188** and the feature **184**B associated with the thumb is filtered out. In alternative arrangements, the thumb feature **184**B may be treated as a separate feature by itself (or in another group), for example, to alter the input or operational mode of the system or to implement another gesture, for example, a slider gesture associated with an equalizer slider displayed on the screen in the area of the thumb (or other finger).

[0091] As shown in **FIG. 6E**, the key parameters of the feature group **188** are calculated for each image **180** and **182**. The key parameters associated with the first image **180** represent the initial state and the key parameters of the second image **182** represent the current state.

[0092] Also as shown in **FIG. 6E**, the knob **170** is the UI element associated with the feature group **188** because of its proximity to the knob **170**. Thereafter, as shown in **FIG. 6F**, the key parameter values of the feature group **188** from each image **180** and **182** are compared to determine the rotation vector, i.e., the group of features rotated five (5) degrees clockwise from the initial to current state. In **FIG. 6F**, the initial feature group (image **180**) is shown in dashed lines while the current feature group (image **182**) is shown in solid lines.

[0093] As shown in **FIG. 6G**, based on the rotation vector the speaker **192** of the tablet PC **175** increases (or decreases) its output in accordance with the amount of rotation of the fingers **176**, i.e., increase the volume by 5% based on rotation of 5 degrees. The display **174** of the tablet PC can also adjust the rotation of the knob **170** in accordance with the amount of rotation of the fingers **176**, i.e., the position of the knob **170** rotates five (5) degrees. In most cases, the rotation of the knob occurs simultaneously with the rotation of the fingers, i.e., for every degree of finger rotation the knob rotates a degree. In essence, the virtual control knob follows the gesture occurring on the screen. Still further, an audio unit **194** of the tablet PC may provide a clicking sound for each unit of rotation, e.g., provide five clicks based on rotation of five degrees. Sill yet further, a haptics unit **196** of the tablet PC **175** may provide a certain amount of vibration or other tactile feedback for each click thereby simulating an actual knob.

[0094] It should be noted that additional gestures can be performed simultaneously with the virtual control knob gesture. For example, more than one virtual control knob can be controlled at the same time using both hands, i.e., one hand for each virtual control knob. Alternatively or additionally, one or more slider bars can be controlled at the same time as the virtual control knob, i.e., one hand operates the virtual control knob, while at least one finger and maybe more than one finger of the opposite hand operates at least one slider and maybe more than one slider bar, e.g., slider bar for each finger.

[0095] It should also be noted that although the embodiment is described using a virtual control knob, in another embodiment, the UI element can be a virtual scroll wheel. As an example, the virtual scroll wheel can mimic an actual scroll wheel such as those described in U.S. Patent Publication Nos: 2003/0076303A1, 2003/0076301A1, 2003/0095096A1, which are all herein incorporated by reference. For example, when the user places their finger on the surface of the virtual scroll wheel and makes a swirling, rotational or tangential gesture motion, a scrolling action can be performed with respect to a list of items displayed in a window.

[0096] **FIG. 7** is a diagram of a touch-based method **200** in accordance with one embodiment of the present invention. The method generally begins at block **202** where a user input that occurs over a multipoint sensing device is detected. The user input includes one or more touch inputs, with each touch input having a unique identifier. Following block **202**, the touch-based method **200** proceeds to block **204** where the user input is classified as a tracking or selection input when the user input includes a single unique identifier (one touch input), or is classified as a gesture input when the user input includes at least two unique identifiers (more than one touch input). If the user input is classified as a tracking input, the touch-based method **200** proceeds to block **206** where tracking is performed corresponding to the user input.

[0097] If the user input is classified as a gesture input, the touch-based method **200** proceeds to block **208** where one or more gesture control actions are performed corresponding the user input. The gesture control actions are based at least in part on changes that occur with or between the at least two unique identifiers.

[0098] **FIG. 8** is a diagram of a touch-based method **250** in accordance with one embodiment of the present invention. The touch-based method **250** generally begins at block **252** where an initial image is captured during an input stroke on a touch sensitive surface. Following block **252**, the touch-based method **250** proceeds to block **254** where the touch mode is determined based on the initial image. For example, if the initial image includes a single unique identifier then the touch mode may correspond to a tracking or selection mode. On the other hand, if the image includes more than one unique identifier, then the touch mode may correspond to a gesture mode. Following block **254**, the touch-based method **250** proceeds to block **256** where a next image is captured during the input stroke on the touch sensitive surface. Images are typically captured sequentially during the stroke and thus the there may be a plurality of images associated with the stroke. Following block **256**, touch-based method **250** proceeds to block **258** where a determination is made as to whether the touch mode changed between capture of the initial image and capture of the next image. If the touch mode changed, the touch-based method **250** proceeds to block **260** where the next image is set as the initial image and thereafter the touch mode is again determined at block **254** based on the new initial image. If the touch mode stayed the same, the touch-based method **250** proceeds to block **262** where the initial and next images are compared and one or more control signals are generated based on the comparison.

[0099] **FIG. 9** is a diagram of a touch-based method **300** in accordance with one embodiment of the present invention. The touch-based method **300** begins at block **302** where a GUI object is output. For example, a processor may instruct a display to display a particular GUI object. Following block **302**, the touch-based method **300** proceeds to block **304** where a gesture input is received over the GUI